

9 - Name Services

Sample:

[5 Marks] Suppose you need to develop a DNS resolver which only supports iterative navigation (starting from the root server) to resolve a given hostname to the corresponding IP address. Write the high-level steps (at most 10 lines) for the DNS resolver.

1. The client first send the hostname to the DNS resolver.
2. The DNS resolver contacts name servers to resolve the hostname.
 - First contact the root server.
 - All name serves either resolve the hostname or suggest to contact another name server.
3. Resolution continues until the hostname is fully resolved or it is not valid.
4. The DNS resolver sends the IP address resolved back to the client.
5. The client can connect to the host server using the IP address.

21S1

Q12.9: Domain Name System is organised as a

- hierarchical system ✓
- centralized system
- master-slave system
- peer-to-peer system

16S2

Q8: (b) [2 marks] Explain how DNS can be used to distribute the load of incoming requests over a set of servers.

In most Linux distributions, **DNS by default sends the list of IP addresses in a different order each time it responds to a new client, using the round-robin method.** As a result, different clients direct their requests to different servers, effectively distributing the load across the server group.

(c) [3 marks] Consider an application where resource objects are stored across a multi-server system. As a requirement of the system, given a name of an object, it is critical that the object needs to be accessible as fast as possible. Describe a name space and name format that would be best to address this requirement and explain why. Is your name format an example of a pure name or not?

Pure name reveals no usable information about the entity it identifies (e.g. MAC address). Use URI.

Tutorial W12

Q1: Define and discuss the concepts of Uniform Resource Identifiers

- Uniform Resource Identifiers (URIs) are concerned with identifying resources on the Web, and other Internet resources such as electronic mailboxes.
- URIs are intended to allow a generic way of specifying the identifier so as to make it easy for common software to process the identifier. This allows new types of identifiers to be readily introduced and for existing identifiers to be used by a wide variety of different software and services.
- URIs provide ways to locate the resource being named. They clearly suffer if the resource has since changed its name (e.g. broken links in the Web).
 - With specifying type of protocol (how) and remote name (where)
- URI, URL & URN
 - URI (uniform resource identifiers) identifies a resource (text document, image file, etc)
 - URL (uniform resource locator) is a subset of the URIs that include a network location
 - URN (uniform resource name) is a subset of URIs that include a name within a given space, but no location

Q2: What are Name Services and why do we need them?

- In a Distributed System, a Naming Service is a specific service whose aim is to provide a **consistent and uniform naming of resources**, thus allowing other programs or services to localize them and obtain the required metadata for interacting with them.
- The major operation of a name service is to **resolve a name** (along with consistent naming scheme), i.e to lookup the attributes that are bound to the name.
- Name management is separated from other services largely because of the openness of distributed systems , which brings the following motivations
 - **Unification:** Resources managed by different services use the same naming scheme, as in the case of URIs.
 - **Integration:** To share resources that were created in different administrative domains requires naming those resources. Without a common naming service, the administrative domains may use entirely different name formats.
- Key benefits
 - Resource localization (resources can be files, webpages, services, databases, etc..)

- Uniform naming
- Device independent address (e.g., you can move domain name/web site from one server to another server seamlessly)
- Naming Services are not only useful to locate resources but also to gather additional information about them such as attributes (its domain, resource type, etc ..).

Q3: What is navigation and what are the approaches to navigation?

- When the name service is distributed then a single server may not be able to resolve the name. The resolve request may need to propagate from one server to another, referred to as navigation.
- Classification of different navigation approaches
 - **Iterative navigation:** The client makes the request at different servers one at a time. The order of servers visited is usually in terms of domain hierarchy . Always starting at the root server would put excessive load on the root.
 - **Multicast navigation:** The client multicasts the request to the group (or a subset) of name servers. Only the server that holds the named request returns a result.
 - **Non-recursive server controlled navigation:** The client sends the request to a server and the server continues on behalf of the client, as above.
 - **Recursive-server controlled navigation:** The client sends the request to a server and the server sends the request to another server (if needed) recursively.

Q4: What is Domain Name System and in what aspects does it improve on a file-based implementation?

- The Domain Name System (DNS) is a name service design whose main naming database is used across the Internet.
- Before DNS, all host names and addresses were held in a single central master file and downloaded by FTP to all computers that required them.
- The problems with the original name service included:
 - It did not scale to large numbers of computers.
 - Local organizations wished to administer their own naming systems.
 - A general name service was needed not one that serves only for looking up computer addresses.
- DNS is designed for use in multiple implementations, each of which may have its own name space, though in practice the Internet DNS name space is the one in widespread use.